Date=19/08/2020
Lecture By=Shubham Joshi
Subject ⇒ Revision of Heaps

IN PREVIOUS LECTURE **(QUICK RECAP) Date-18/08/2020**	**In Today's Lecture (Overview)**
⇒ Heaps in Python ⇒ MCQS ⇒ Questions for self Practice // CC for the day	Heapify in Heap Types of Heaps Minimum Heap Maximum Heap Questions For Self Practice // Assignment for the Day

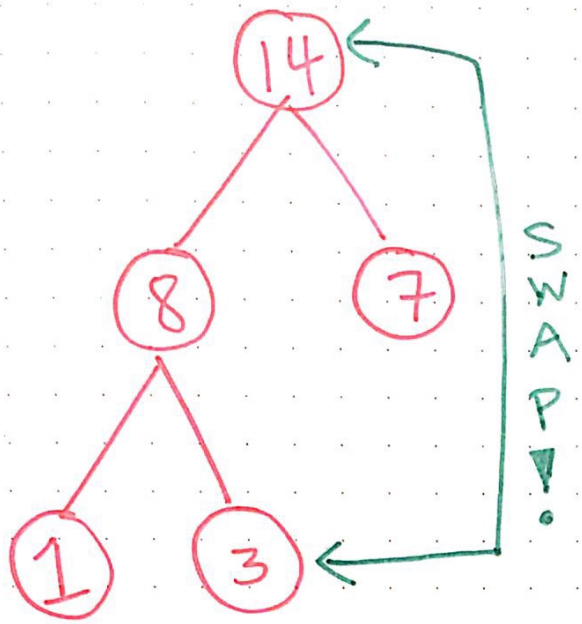
Heapify in Heap

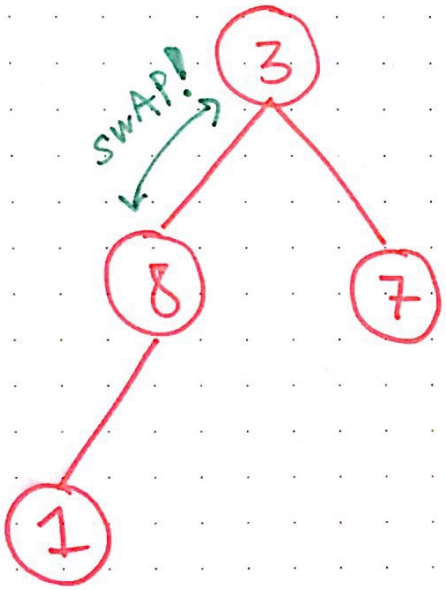
Heapify is the process of converting a binary tree into a Heap data structure. A binary tree being a tree data structure where each node has at most two child nodes. A Heap must be a complete binary tree, that is each level of the tree is completely filled

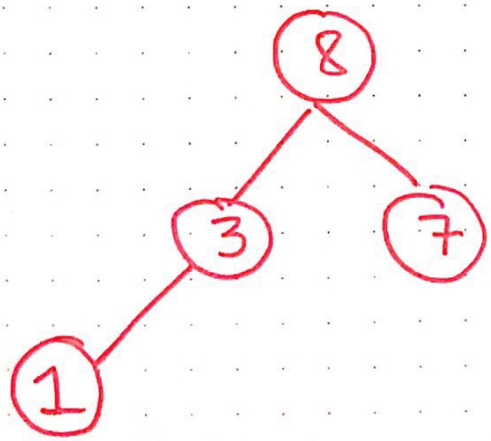
This function converts a regular list to a heap.

In the resulting heap the smallest element gets pushed to the index position 0.

But the rest of the data elements are not necessarily sorted.





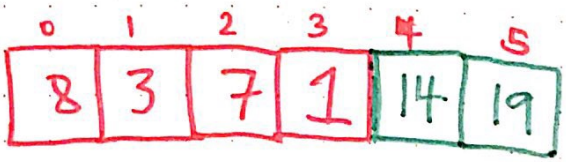


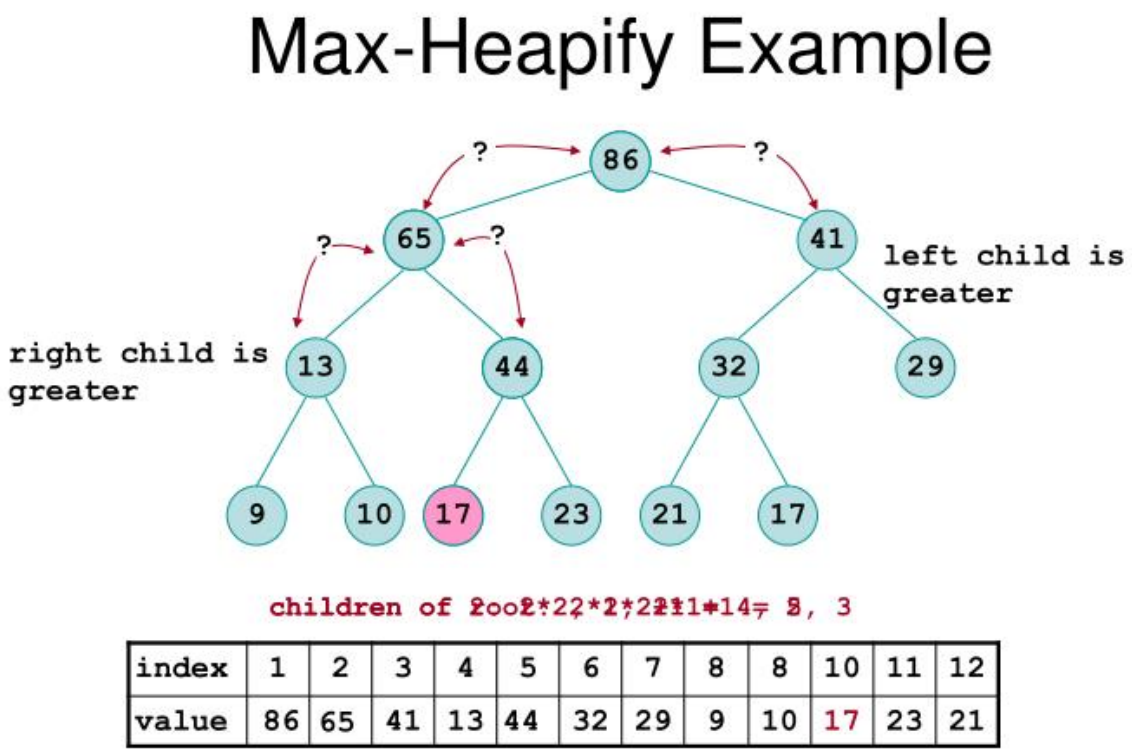
* Once we are back in a max heap state, we can continue repeating the same steps until we are left with a heap size of 1:

→ swap first + last elements.

→ remove last node as it is already in its sorted position in the array.

→ heapify until back at a max heap state.





children of 2oo2*22*2;22*1*14= 3, 3

index	1	2	3	4	5	6	7	8	8	10	11	12
value	86	65	41	13	44	32	29	9	10	17	23	21

Types of Heaps

Minimum Heap

A Min-heap is a complete binary tree in which the value in each internal node is smaller than or equal to the values in the children of that node.

Mapping the elements of a heap into an array is trivial: if a node is stored a index k, then its left child is stored at index **2k + 1** and its right child at index **2k + 2**.

Example of Min Heap :

```
           5                      13
         /      \               /       \
       10        15            16         31
      /                       /  \        /  \
    30                     41     51    100   41
```

How is Min Heap represented ?

A Min Heap is a Complete Binary Tree. A Min heap is typically represented as an array. The root element will be at **Arr[0]**. For any ith node, i.e., **Arr[i]**:

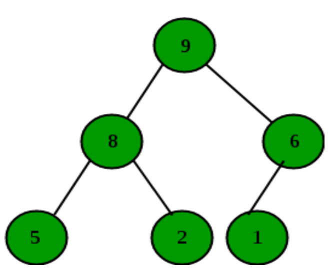
- **Arr[(i -1) / 2]** returns its parent node.
- **Arr[(2 * i) + 1]** returns its left child node.
- **Arr[(2 * i) + 2]** returns its right child node.

Maximum Heap

A Max-heap is a complete binary tree in which the value in each internal node is greater than or equal to the values in the children of that node.

Mapping the elements of a heap into an array is trivial: if a node is stored a index k, then its left child is stored at index **2k + 1** and its right child at index **2k + 2**.

Examples of Max Heap :



How is Max Heap represented ?

A Max Heap is a Complete Binary Tree. A Max heap is typically represented as an array. The root element will be at Arr[0]. Below table shows indexes of other nodes for the ith node, i.e., Arr[i]:

Arr[(i-1)/2] Returns the parent node.

Arr[(2*i)+1] Returns the left child node.

Arr[(2*i)+2] Returns the right child node.

Questions For Self Practice // Assignment for the Day

Q 1-> Design a heap with Heapify method